DOCUMENT RESUME

ED 195 231 IR 008 936

AUTHOR Theis, Douglas J.
TITLE Distributed Network and Multiprocessing Minicomputer State-of-the-Art Capabilities.
PUB DATE 80
NOTE 13p.: Paper presented at the National Computer Conference (Norfolk, VA, June 23-25, 1980).

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Comparative Analysis; *Computers; Equipment Standards; *Minicomputers; Selection

ABSTRACT

An examination of the capabilities of minicomputers and midicomputers now on the market reveals two basic items which users should evaluate when selecting computers for their own applications: distributed networking systems and multiprocessing architectures. Variables which should be considered in evaluating a distributed networking system capability include topology, line protocols, hardware interfaces, network protocol, user software, and network software. With regard to the multiprocessing architectures capability, minicomputer and midicomputer vendors provide several different kinds of capabilities for interconnecting multiple computers into multiprocessing systems. Four manufacturers offer bus interconnected hardware between CPU's, two vendors supply turn key systems, and six offer shared main memory products; most vendors offer shared disk hardware. Since each user has different needs, it would be difficult to establish an overall rating system for deciding upon a particular system; however, key performance advantages and disadvantages of each type of capability have been identified. (Author/LLS)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED1952231

U S DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

DISTRIBUTED NETWORK AND MULTIPROCESSING MINICOMPUTER STATE-OF-THE-ART CAPABILITIES

by

Douglas J. Theis

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Douglas J. Theis

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

IR008936

Distributed network and multiprocessing minicomputer state-of-the-art capabilities

by DOUGLAS J. THEIS
The Aerospace Corporation
El Segundo, California

Today's minicomputer and midicomputer state-of-the-art provides two basic types of capabilities for users to evaluate for their applications. The two categories are:

- Distributed Networking Systems
- Multiprocessing Architectures with High Speed Busses, Shared Main Memory and Shared Disk Systems

A distributed network system (e.g., Digital Equipment Corporation's DECNET, Hewlett Packard's DSN, Modcomp's MAXNET) offers standard data communications interfaces with line protocols over both telephone lines and hardwired cables. On the other hand, multiprocessor approaches offer several interconnection schemes. An interprocessor bus (e.g., Data General's Multiprocessor Communication Adapter and Digital Equipment's Parallel Communication Link) can provide high data transfer rates among processors, with either custom user-developed software or a vendor developed real-time operating system. This bus interconnect approach is also evolving into configuration implementations with a significant amount of fail-soft or graceful degradation capability (e.g., Tandem's minicomputer and BTI's midicomputer system) and a turn key operating system that fully supports multiprocessing.

Practical multiprocessing is also achieved by sharing memory of some sort so that many CPUs can access one common memory. Shared main memory provides the necessary multiported capability so that two or more CPUs can access one common memory. This means that special instructions and software are required for a well disciplined method of resolving contention and simultaneous memory request conflicts. The shared memory multiprocessing can also be achieved with a shared disk unit, where two or more computers have access to one common disk. Shared disk provides more memory capacity, but the time to read or write data is longer. Most major minicomputer and midicomputer manufacturers offer shared disk hardware capability. Table I gives a status on minicomputer and midicomputer vendors which support both distributed networking and multiprocessing capabilities.

DISTRIBUTED NETWORKING MINI/ MIDICOMPUTER SYSTEMS

Distributed network processing is definitely upon us and being used in many applications, but the availability of off-the-shelf, turn key distributed networking systems has been limited. Although the necessary hardware, such as data communication controllers and interfaces, has been available, the constraining force has been the development of the software. The total objective is to make the hardware links and message/communication protocols transparent to the user; as a result each vendor has had to make many design decisions about the network control methods and access methods. All this is needed in order for the user to have higher-order level language statements for network operation using just logical references, which in turn do the remote operations with other nodes in the distributed network. Even now only some of the major vendors offer substantial support, but just about all the others are developing this capability to catch up with the obvious trend.

Each manufacturer has started out offering partial capabilities because it is really not practical to develop everything one could think of as desirable. For example, Hewlett Packard's first network system, which interconnected distributed minicomputers throughout their own factory, was designed to provide peripheral sharing and central programming development services. So far each vendor has gone about providing distributed network capabilities in a somewhat different manner, yet providing means to accomplish similar types of end objectives. It is difficult to compare vendors' offerings of distributed network capabilities for the very reason that there are so many different ways to accomplish similar top-level operations.

There are at least two dimensions to a distributed network mini/midicomputer system: one dimension is the network configuration (i.e., topology of nodes) and the other is the layers within each node, from the bottom layer of the electrical interface to the top layer consisting of the network access method, which allows the user complete transparency of the network characteristics. The major topics discussed

1080

TABLE I—Mini/midicomputer vendor support for four configuration types

Manufacturer/ Model Nos.	Type 1 Distributed Network	Type 2 Multiple CPUs Interconnected With Bus Hardware	Type 3 Shared Main Memory	Type 4 Shared Disk
BTI Computer Systems, Inc. BTI 800		x		x
Computer Automation Inc. LSI 4/10, 4/30, 4/90	x			x
Control Data Corp. Cyber 18 Series, MP80	x			x
Data General Corp. Nova and Eclipse Series	x	x		x
Digital Equipment PDP 11 Series VAX 11/780	x x	x	x	x x
General Automation 440, 460, 440DS, 460NDS, 550	x			x
Harris Computer Systems Slash 4, 6, and 7 Series			x	x
Hewlett Packard 1000 and 3000 Series	x			
IBM 8100 Series	x			
Modular Computer Modcomp II, IV, and Classic Series	x		x	x
Perkin-Elmer (formerly Interdata) 7/32, 8/32, 3220, 3240			x	x
Prime Computer P350, 450, 550, 650, and 750 Series	x			x
System Engineering Laboratories, Inc. 32/35, 32/55, 32/75			x	x
Tandem Computers T16 Series	x	x		x
Texas Instruments 960, 980, 990				x
Univac (formerly Varian) 77-400, 77-600, 7-800	x		x	x

within this distributed network section are as follows:

- Topology
- Line Protocols
- Hardware Interfaces
- Network Protocol
- User Software
- Network Software

The user learning about distributed network capabilities quickly finds many new acronyms and terminology related to this industry. Some of the more common ones are presented in Table II.

Topology

Network configuration, or topology, refers to the interconnection pattern of nodes and links to form the network. The most commonly used configuration of networks is point-to-point, where each node has its own links to other nodes. Multidrop is the other type of configuration (sometimes called multipoint); it is different in that two or more nodes time share a link to save on the communication link costs. Point-to-point capabilities are supported by all the Type 1 vendors presented in Table I. Multidrop was available from some vendors in 1979 and will be supported by others sometime in 1980. Many point-to-point configurations are often found in hierarchical networks where the top level is the host or central node (as in a star network) with the second or bottom level being satellite nodes collecting and processing data in a distributed fashion. Differing topologies are exemplified by Digital Equipment Corporation's DECNET which supports any point-to-point network combination, and by IBM's 8100 which emphasizes a loop or ring structure topology locally. Any distributed network topology has the usual address field or software table limits as to the maximum number of nodes, links, device status, number of paths, etc. which is part of any finite system, but these can be extended when necessary both in size and for more complex applications.

Line protocols

The communication line protocols (i.e. link level control procedures) used over these circuits are some of the most confusing and difficult aspects in distributed network computer systems. During the 1960's remote batch applications

TABLE II—Distributed network related terminology/definitions

BASIC TERMS

Node - can be a terminal or computer termination point which has the capability to transmit and receive over communication line facilities; usually an addressable entity.

Link - the external circuit connection to provide a means of exchanging data via telephone, hardwired cable, or private communication facilities.

Topology - the interconnection of nodes and links into a network configuration for overall operations of one kind or another.

Protocol - a formal set of rules for specifying the format and relationships of message exchanges among communicating nodes.

NETWORKING CONFIGURATIONS

Point-to-point - a network configuration or topology between two nodes with a direct link (i.e. two nodes are connected without any intermediate nodes in between).

Multidrop - a communication link on which two or more nodes (computer or terminal) can share access. This type of link typically requires both polling (or a reservation scheme or contention) and addressing techniques to facilitate shared link operation.

Loop or Ring - a ring like network configuration where each computer node is connected to two adjacent nodes forming a closed loop.

NETWORK SWITCHING

Circuit switching - an electrical connection between two nodes (i.e. over a communication link) is established on demand for dedicated bandwidth of that circuit (i.e. link) until it is dropped.

Message switching - is a capability for messages to be moved over many intermediate links and nodes because the destination address is integral to the message to allow variable paths to be taken as availability permits.

Packet Switching - is a subset of message switching where messages are broken into packets including their own necessary address and control as to their destination. Packet switching allows efficient use of network resources because it breaks up data into a relatively small size (e.g. X.25 allows up to 256 bytes) and no long term storage is required at the various routing nodes.

Virtual Circuits - is a logical circuit connection between two nodes in a network which can be realized with different circuits during the transmission of a message. This service is guaranteed, sequential related operations including end-to-end flow control.

Datagrams - is a capability which provides link communication on non-related (may or may not be guaranteed), non-sequential operations basis primarily toward transaction oriented users where the users provide end-to-end integrity.

and terminals used character oriented protocols such as IBM's well-known BISYNC (i.e. BSC) and others. During the 1970's, the trend toward transaction-oriented systems drove the industry into the era of new protocols such as IBM's SDLC. Also DEC has its DDCMP byte protocol which is designed using a count field instead of a unique flag character as is done in the SDLC bit protocol. Thus bit stuffing protocols (e.g., IBM's SDLC) versus DEC's imbedded character protocol is a controversial subject in that there are trade-offs associated with each. Table III shows a comparison of the communication line protocol presently supported by mini/midicomputer manufactureres as well as BISYNC which is still used for reasons of compatibility with the older remote batch terminal and display terminal equipments. In Table III there is a breakdown of some specific capabilities each of the five protocols supports. There are many other factors beside these to consider in the area of link protocols, but it is beyond the scope or overview we are trying to present in this article.

One new trend that is certain to be offered by more vendors is the use of an integral microcomputer to do the line protocol functions and thus decrease the amount of code and overhead in the node distributed network software subsystem. In particular Digital Equipment Corporation offers two types of microprocessors: (1) DMC-11 with a ROM memory implementing their DDCMP protocol and (2) KMC-11 with a RAM memory so any other protocol (e.g. IBM's) can be implemented. Note that the Signetic's protocol chip advertised for doing the DDCMP protocol does the framing function which comprises only a part of the operations for the DDCMP protocol. In the case of Hewlett Packard, their line protocol software is implemented in the node minicomputer microcode/control store memory. This off-loads part of the Hewlett Packard Distributed System Network software resident in main memory. Hewlett Packard plans to use an integral microcomputer in their next DSN upgrade. These types of approaches should improve throughput performance.

Hardware interfaces

Distributed network computer system products are new systems and as such high "effective user" data rate throughputs are difficult to achieve until after functional software design and operational experience has matured. The throughput inefficiencies are primarily in the software overhead features developed to make system network details

TABLE III—Line protocol features and characteristics

FEATURE	BISYNC	DDCMP	ADCCP	HDLC	SDLC
Protocol Type (i.e. Data Transparency)	Character Stuffing	Count	Bit Stuffing	Bit Stuffing	Bit Stuffing
Full Duplex	No (HD)	Yes	Yes	Yes	Yes
Serial	Yes	Yes	Yes	Yes	Yes
Parallel	No	Yes	No	No	No
Multi-Acknowledge by one ACK	No	Yes	Yes	Yes	Yes
Asynchronous Operation	No	Yes	No	No	No
Point-to-Point	Yes	Yes	Yes	Yes	Yes
Multipoint	Yes	Yes	Yes	Yes	Yes
Error Detection (CRC)	CRC-16	CRC-16	CRC-CCITT	CRC-CCITT	CRC-CCITT
Retransmit Error Recovery	Yes	Yes	Yes	Yes	Yes
Count Field Required	No	Yes	No	No	No
Protocol Overhead (single data message)	80 bits and DLE additions	80 bits	48 bits fixed and O's for bit stuffing	48 bits fixed and O's for bit stuffing	48 bits fixed and O"s for bit stuffing

transparent to users. Then the more network support software that is provided to do all these nice features drives up the complexities which need to be refined, tuned and optimized to maximize throughput performance. This is easy to say but obviously requires the appropriate software/hardware trade-offs of how much complexity is really necessary and some smart design implementations to achieve the highest throughput performance possible.

The physical links between nodes are either data communication lines, with modems and adapters attached to the computer channels, or hardwired cables. At the device level, the EIA (Electronics Industries Association) RS-232-C standard is provided by all the vendors. Other standard circuit connections that are either supported or planned to be supported include the following: Electronics Industries Association EIA RS-422 and RS-423, Western Electric 301/303, MIL-STD-188C and the new Opto-Isolated fiber optics interfaces. RS-232-C provides operation up to 19,200 bits per second. Higher data rates and better performance can be accommodated by the newer EIA RS-422 (up to 10M bps) and the EIA RS-423 (up to 100K bps) circuit standards. The Western Electric 301/303, supported by several vendors such as Modcomp, permits data rates from 19,200 bits per second up to 230,400 bits per second over special leased circuits. MIL-STD-188C is very similar to RS-232-C and vendors can support it in special situations as needed for the government and the military. So far standards for opto-isolated interface have not been developed to the same level as the above electrical interface standards.

The vendors in Table I support many but not all of the commercially available data link services from voice grade telephone lines up to 9,600 bits per second, leased (private) lines up to 230.4K bits per second, DDS (Dataphone Digital Service) up to 56,000 bits per second, Bell's T1 Carrier and Satellite Carriers (e.g. Western Union's Weststar) both up to 1.544M bits per second. Presently Digital Equipment Corporation's DECNET II using their DDCMP protocol has hardware interface and software drivers to accommodate 9,600, 19.2K, 250K, and 1M bits per second for point-to-point hook-ups using the appropriate modems or special cable interfaces. For multidrop arrangements, Digital Equipment Corporation provides both the low speed 9600/4800 (outbound/inbound) bits per second with modems and a high speed, 512K bits per second, synchronous, full duplex capability over cables. Hewlett Packard's DSN in point-to-point networks accommodates up to 56,000 bits per second with modems and up to 480,000 bits per second over cables. DSN supports both the low speed 9600/4800 (outbound/inbound) multidrop configuration as well as a 19,200 bits per second multidrop capability. Modcomp's MAXNET supports point-to-point configurations up to 19,200 bits per second with RS232-C compatible modems, 40.8K with WE 301 modems, and 19.2K, 50K, 230.4K with WE 303 modems. Without modems in a point-to-point configuration, MAXNET supports either 19,530 bits per second or 1.25M bits per second using cables and internal clock. MAXNET supports multidrop using RS232-C modems capable of multidrop up to a maximum of 9,600 bits per second outbound and 4,800 bits per second inbound. Modcomp also supports 250K and 500K bits per second multidrop for up to 5,000 feet with a limit of 8 drops (distance is dependent on speed and number of drops).

Network protocol

There is a network standard and that is the CCITT (International Consultative Committee for Telegraphy and Telephony) X.25 network access protocol. It is a framework within which to build a system that allows vendors to implement their own efficient software designs as well as inclusion of many additional or optional features. Several vendors have already begun to implement a compatible version for their own distributed network systems to provide the necessary control functions, message formats and other capabilities necessary to implement the compatible node to node operations so that the user can do normal network access operations with other X.25 networks. The X.25 Network Access Protocol was developed by CCITT Study Group VII in July 1976 (revised April 1977) and is supported by Telenet, Datapac (Canada), Euronet (Europe), and others. The X.25 protocol consists of three access levels:

1. Level 1 defines the physical and line interfaces used (i.e., RS-232-C).
2. Level 2 defines the link level procedures in terms of message formats and sequencing (i.e., HDLC).
3. Level 3 defines the packet format used to establish and exchange data (i.e., virtual circuit).

For example, Digital Equipment has under development an X.25 compatible interface for its own DECNET so user tasks and DECNET utilities can communicate with a remote computer node over any public packet switching network using X.25 protocol. Two program interfaces are needed to provide a common basis for call, set-up, call disconnect and exchange of data between DECNET machines and other types over a X.25 network. Hewlett Packard plans to interface to these public X.25 networks and Modcomp also plans to have X.25 software interface compatibility in the not too distant future. Table IV provides information on each vendor's network access protocol/layers of software to show the overall framework and its related terminology.

At least one manufacturer has announced a chip (i.e., Western Digital's 2501) that is specifically designed to replace the software required for the first two levels (physical and link access) of the X.25 packet switching protocol. The VSLI chip has approximately 30,000 gates. Maximum data rates of up to 1.3M bits per second are planned to be accommodated by this chip. When level three (network control) is firmly specified, then it will also be put on a chip. In addition, this chip design provides link and loop back testing and has a programmable address field to allow terminal intercommunications. Prior to the X.25 being on one or a few chips, it was too expensive to make intelligent terminals as nodes in a public network and now that trend is changing. Note, however, that terminal networks is not the subject of this article.

TABLE IV—Manufacturer's network software layer/protocol structure

Manufacturer	Network Architecture Acronym/ Designation	Mini/Midi Computer Models Supported	User Program Interface	Network Link Level	Message Protocol (e.g. SDLC)	Link Hardware Interface (e.g. RS 232-C)
Computer Automation Inc.	SyFA (System For Access) VN (Virtual Network)	LS2 4/10, 4/30, 4/9[illegible]	Not Supported	Emulates IBM 3270, 3780, others	Bisync	RS 232-C
Control Data Corporation	NAM-18 (Network Access Method)	Cyber 18 Series	Under development	X.25 Compatible under development	Mode 4, CDCCP (ADCCP, HDLC, and SDLC)	RS 232-C
Data General Corp.	None, not developed	Nova's and Eclipses	Not Supported	X.25 Supported	HDLC	RS 232-C
Digital Equipment Corporation	DNA (Digital Network Architecture)	PDP 11 Series VAX 11/780	Vendor offers several: DAP, ATS, User	Vendor calls Network Service Protocol (NSP)	DDCMP	RS 232-C WE301/303 CCITT V.35
General Automation Inc.	AUTONET	Models 220, 440, 440DS, 440 NDS, and 550	Under Development	X.25 Compatible under Development	ADCCP, SDLC	RS 232-C
Hewlett Packard Inc.	DSN (Distributed System Network)	1000 Series 3000 Series	Vendor calls Network Access Method	Vendor calls Network Manager DS/1000 DS/3000	ADCCP, HDLC	RS 232-C WE 301/303 CCITT V.35
IBM	SNA (System Network Architecture)	8100 Series	Function Management	Transmission Control & Path Control (2 layers)	SDLC	RS 232-C
Modcomp Inc.	MAXNET	Modcomp II, IV and Classic Series	Vendor calls REX	Vendor unique	ADCCP, HDLC, SDLC	RS 232-C WE 301/303 CCITT V.35
Prime Computer Inc.	PRIMENET	350, 400, 450, 500, 550, 650, and 750	Inter Program Communications Facility (IPCF), File Access Manager (FAM)	Packet Network Interface (PNI) X.25 Compatible	HDLC	RS 232-C
Tandem Computers	GUARDIAN/ EXPAND	Tandem 16 Series	Vendor Unique	X.25 Compatible under development	HDLC	RS 232-C
Univac (formerly Variat)	DCA (Distributed Communications Architecture)	77-400, 77-600 77-800	GRAM (Global Resource Access Module)	Vendor calls Transport Network System	DDLC handle HDLC and SDLC as subsets)	RS-232-C

The network control software module layer (typically level 3) below the user's level takes their requests and decodes them as such into link and node assignments. These assignments are allocated at system generation time so the necessary reformatting and control is already established to carry out the desired operation. There is no standard because each vendor defines and implements it to suit himself. Some vendors have more layers in their software network structures than others. This software does the routing, sets the control fields according to prescribed methods, and performs any other functions which are necessary to provide desired cperations at another node. The network control software can be designed to provide alternate routing as well as other items related to network transfers. When there are more than two levels of nodes (i.e. intermediate nodes between sending and receiving nodes), some degree of routing capability is needed. There are many and varying degrees of capability one can implement for network routing. One basic capability is to provide automatic rerouting when a node fails, but it may require the user to reinitiate the job. The objective is to make rerouting or changing link paths transparent to the user. Dual path capability between nodes with some degree of line load sharing to maximize link utilization is an additional complexity to provide in network routing. Vendors start out offering some basic routing capabilities and then evolve into more complex routing capabilities as time goes on. It is well to remember that routing capabilities can vary from some basic features all the way up to some very sophisticated ones.

User software

At the user or application level there are four types of general purpose higher order level language/command capabilities needed for basic network operations. Here are some typical capabilities of distributed network services available from Type 1 vendors in Table I:

1. Remote file access and transfer—gives user full access to data files at any remote node in the network.
2. Downline loading—provides a host node the capability to downline load code to any satellite node for later use.
3. Remote command processing—allows user at any node to commence task execution on any other node in the network with similar set of operating system commands.
4. Program to program communication—the capability allows user application programs at one node to exchange data and control information with any program at another node. Each vendor has variations and supplementary provisions to use this capability in different ways such as equal partners which are dynamically variable or in master/slave arrangements.

There are many different ways to do these network functions between nodes but basically they consist of either file transfer or program-to-program operations. The particular way network operations are done depends on the application and/or user discretion. In the typical data reduction network scenario the user usually has the computer nearest the data storage (not necessarily the data acquisition computer) execute the program for data reduction. For example if a large data base needs to be searched on the host node to do some satellite node processing, then the program-to-program communication approach would logically minimize link traffic in having the host do the processing and later pass over the results to the satellite node. The reverse is also true; when large numbers of files (stored at the host node) are needed for local (or satellite) node processing, one would typically use remote file access and transfer files for use.

Remote command processing provides the much advertised advantage of distributed processing, namely, peripheral resource sharing. Each vendor implements his network software to provide the user with some degree of peripheral device transparency and/or device independent operations.

User transparency for tape and disk storage operations is fully supported but for other devices such as printers and displays, this may or may not be practical or desirable depending on the application.

Downline loading from a host node to a satellite node is the other obviously needed capability for any distributed network software. The two advantages for this are related to minimizing the number of development resources and minimizing equipment at the satellite node. For instance, only the code required resides in the satellite node but it can be changed and reloaded as often as necessary. There are several types or degrees of capability one might like in downline loading control and execution of jobs on remote nodes. Basically then, downline loading can move the system and user jobs to the satellite and start execution of same. An additional capability offered by DECNET, for instance, is to transmit an additional job to the satellite node while the previous jobs are running and have that new job start in its execution also on the satellite. So far no vendor supports overlay programs on the satellite from the host.

Network software

Distributed network node software is designed and implemented in a modular and layered fashion to facilitate added capabilities (i.e. growth and flexibility), and thus minimize impacts on modules or layers not directly changed. In a real sense it's a structured programming methodology. For obvious product considerations, the distributed network software functions are a separate yet compatible extension of the vendor's proven operating systems. Thus a user loads this distributed network software subsystem (usually designated by some clever acronym) in addition to or in conjunction with the operating system. From this point, each vendor develops his own implementation details for the distributed network software package.

Each Type 1 vendor in Table I offers special or unique features related to his equipment capabilities and those things found to be useful or necessary in many applications. An example of this is that Digital Equipment Corp. sells machines of 12, 16, 32 and 36 bit words and in order to network any combination of these they offer standard supported interface hardware and software to make these differences transparent to the user under DECNET. Hewlett Packard's Distributed Systems Network provides "store and forward" storage capability at each node. This facilitates routing and rerouting through the links if any link or node goes down. Modcomp has developed their networking system to improve performance by providing input-output operations concurrent with system operations using special (i.e. Modcomp calls "symbiont") software, and by implementing a "core device" option for direct CPU-CPU transfers which provides a much faster method than normal. Note that none of the first three vendors (DEC, HP, and Modcomp) built its network software for X.25 compatibility, but all three plan to develop compatible interfaces to their software for X.25 operations.

MULTIPROCESSING ARCHITECTURES

Several minicomputer and midicomputer vendors provide several different kinds of capability for interconnecting multiple computers into multiprocessing systems. There are four Type 2 manufacturers offering bus interconnected hardware between CPUs. For Data General and Digital Equipment Corporation minicomputers, the user tailors the vendor's realtime operating system to his application using this hardware. The other two vendors supply turn key systems. Shared main memory products are offered by six vendors as presented in Table I. Table I, Type 4 presents the vendors offering shared disk hardware which just about includes all the vendors with this type of product. The user is responsible for operating system software changes to accommodate their application when using either shared memory or shared disks configurations.

Multiple computer bus interconnected hardware

Four manufacturers provide the computer hardware bus interconnected capability, and the features of these systems are discussed below. Using these hardware capabilities, as is, allows the user to achieve high performance utilization because the system can be tailored to their application. It also provides a means to tie together many computers in multiprocessor configurations with no special hardware development.

Data General was the first major minicomputer manufacturer to offer standard bus hardware for interconnecting up to fifteen machines for multiprocessing operations. Their standard interconnection hardware is designated as the Multiprocessor Communication Adapter (MCA) and was first introduced as a standard product back in 1970. It accommodates any of the Data General Nova or Eclipse line of minicomputers. The MCA in effect connects the direct memory access channels into a ring or daisy chain structure so data transfers can be made from any one CPU's memory to any other designated memory at very fast transfer rates.

Data General recently (1978) announced an additional type of bus interconnection capability called the Multiprocessor Communications System (MCS) which connects up to fifteen of their Novas and Eclipses into a star or radial bus network. It has the same data transfer rating of 1M bytes per second through direct memory channel interfaces. Buffering in provided internal to the MCS and special instructions control data transfers with variable sized block transfer capability. Also computers can be added or removed from the network without affecting overall operation.

Data general

One user implemented a 4 millisecond closed loop missile simulation using three Data General minicomputers in a MCA ring with their Model M600 modeling the missile, their Model S-230 simulating the target, and their Model S-130 being the signal generator control as shown in Figure 1. In

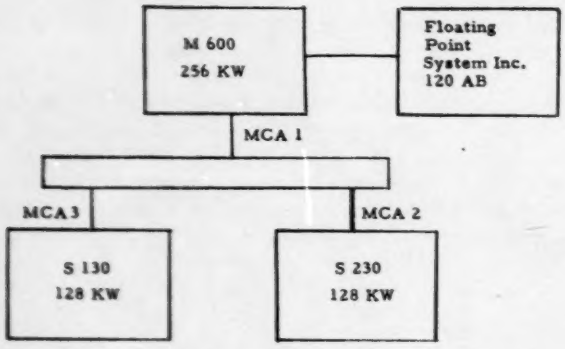


Figure 1—Computer simulator configuration using Data General's MCA bus

order to minimize software overhead on the bus and thus maximize throughput, this user coded his own MCA software drivers and attained an effective transfer rate of 100K to 150K words per second. This is the useful data rate range after overhead, contention, etc., on the MCA bus. All the software development and simulations are run under AOS 2.1, Data General's Operating System. As long as there is sufficient main memory capacity, the MCA bus seems more than adequate to maximize overall throughput.

Digital equipment

Digital Equipment Corporation offers a Parallel Communication Link (PCL11-B) hardware option for ganging together up to 16 PDP 11 minicomputers with a common time shared bus. The PCL11-B hardware attaches to Digital's PDP 11 Series Unibus and if any PDP-11 does down, the others are not affected. User programs logically reference other CPUs on the PCL bus in a fashion similar to designating any peripheral device and PCL is supported under DECNET. Contention for the link is resolved by assigning a time slice to each PCL11 slot. A recent example utilizing this equipment is the Defense Communications Agency's Autodin II, a general purpose data communications network system. Autodin II operates in a packet-switching manner very similar to ARPANET. This multiple computer configuration (i.e., 11/70s) provides high availability by a special configuration which allows graceful degradation. The Autodin II network has eight nodes, each of which consists of 3 to 5 PDP 11/70s tied together with the PCL-11.

Two new multiple CPU, bus interconnected architectures

A more recent trend in multiple CPU, bus interconnected hardware, is offered by two relatively new manufacturers (BTI Computer System and Tandem) which provide complete off-the-shelf user supported software with significant fail-soft capabilities. The manufacturer has to build into the operating system many features to accommodate the many different hardware malfunctions and resulting reassignment of resources. This makes it difficult for each user of such a system to customize his operating system and still maintain the software integrity to support fail-soft operations. In other words, the user is encouraged to do everything in higher order level languages such as FORTRAN, COBOL, BASIC, PASCAL, others, and leave the manufacturer's operating system as is except for well-defined improvements that don't impact the fail-soft operation. These are new architecture implementations, each different, but offering capabilities every user should know.

BTI 8000

The multiprocessing architecture approach taken by BTI Computer Systems Inc. in their new 32 bit machine, Model 8000, centers around their very high speed 32 bit bus (patented) which supports a 60M bytes per second bandwidth. Almost any combination of Central (i.e., computational) Processing Units, Memory Control Units, and Peripheral Processing Units (PPU) can be used in the fifteen available chassis slots (16th slot is required for the System Service Unit). For instance, if the user application is processor bound, then up to thirteen Central Processing Units could possibly be used. In input-output bound cases more Peripheral Processing Units could be installed. Some combination to match the user's loading can easily be accommodated using this highly flexible multiprocessing operation. The key point is that the 60M bytes per second bus should never be the bottleneck or the limiting part of this multiprocessor design. In a time sharing system, one PPU can support up to 256 user terminals each operating at speeds of up to 19.2K bits per second. The other significant aspect of interest in this architecture is that it supports true homogeneous multiprocessing operation where all resources (i.e., CPUs) are equal partners with no master-slave internal to the design. The designers of this system used PASCAL as the basis for much of the computer hardware design so the influence of PASCAL operations has driven much of the internal hardware design features as was originally done in the case of ALGOL for the Burroughs 5500.

The System Service Unit is strictly for the customer engineer to facilitate troubleshooting and repair of the entire BTI Model 8000. It has a small microcomputer (as do the MCU and PPU) which runs the diagnostics to determine faults and isolate them to the chip level. The SSU has its own telephone port for remote troubleshooting from the factory. After any major module goes down it is taken off-line; the user simply does a SYSGEN automatically by just pressing a button.

Tandem T16 series

Tandem Computers Inc. installed its first multiprocessor system back in 1976, and last year the sales of this product were over $25M with projections as high as $40M this year. This noteworthy success of the Tandem T16 Series computers shows that up to sixteen tightly coupled minicomputers can replace a large scale computer with the equivalent

performance and offer additional advantages such as higher availability and more efficient accommodation of large numbers of disks and terminals. The Tandem Computer architecture consists of a high bandwidth (i.e., 26M bytes/sec) bus called Dynabus which provides interprocessor transfers separate from all the peripheral (i.e., disks, tape, terminals, etc.) transfers. These are done over their own direct memory access channel at speeds up to 4M bytes/sec. Interprocessor communication is provided with Dynabus; therefore, no shared main memory is required. The other key multiprocessing architecture success is their operating system (130K bytes per minicomputer processor) that provides complete user transparency to resource control and allocation, load balancing, fail-over, and many other functions.

For example, one Tandem Five processor system is used as a back-end file manager in Columbus, Ohio by the Ohio College Library Corp. It does library catalog processing in conjunction with forty 240 MB disk drives supporting several large host computers (i.e., Sigma 9s). Another Tandem dual processor front-end system is used to interface and operate the 2000 CRT alphanumeric terminals (each with a maximum 2 second response time limit) where each processor has 512K bytes of its own memory. Typically, if the system has a large number of page faults, the processors need more memory and 2M bytes is the maximum per processor. The multiprogramming environment is facilitated with stack oriented CPU design.

The Tandem non-stop or fault-tolerant capability consists of a wide variety of special hardware and software techniques. A basic system has two or more processors, two cable Dynabus peripherals with dual port controllers, and disk drives with dual access control as well as redundant disk drives. Tandem's specially designed disk controller has a 4K byte buffer to support recording to one or more disk drives without having to retransmit on the input-output bus. It also provides the capability for one disk unit to be copied to another without involving the input-output bus. Of course, there is a 10-30 percent throughput performance overhead to provide checkpointing and the necessary updating of file generation in pairs. Any operational module can be replaced on line without stopping operation of the Tandem System. There is also a dual power distribution in Tandem so if one power supply fails, the backup is automatically switched in.

Shared main memory operations

Shared main memory provides capability for multiple CPUs to be connected to one shared memory. The advantage is that each CPU has access to the data in shared memory and typically users do not store executable code in shared memory. Shared memory addresses are established at SYSGEN time typically as Global Common areas in the Fortran sense although it should be true for any language. The linking loader is used to connect logical names to physical addresses so user applications can use data in shared memory for those processing programs which are executed in local (i.e., not shared) memory. The beauty is that all the CPUs with a port on the shared memory module can access that data at very fast main memory access speeds in a well regulated or disciplined method so as to minimize impact on overall processing throughput and also eliminate any need for multiple copies of data residing in each memory of every CPU needing the data for processing. With the cost of memory significantly decreasing all the time, the major benefit is definitely in achieving higher throughput with parallel processing. It does this by minimizing data transfers amongst multiple CPUs because all the CPUs have the necessary common data access.

Perkin-Elmer

As an example, let's go through a typical application as to how a user really benefits from shared main memory capability. Assume the processing algorithms in each CPU take much longer to execute than the time for the buffer shared memory to fill from the source via the DMA channel. In order for the processing to keep up, multiple CPUs must have access to the data buffer (as shown in Figure 2). Without shared memory copies of the same data have to be stored in several memories so processing can be done in parallel with the resulting overall throughput being slower. Figure 2 shows data coming from some source in short bursts at very high speed data rates over a programmably switched direct memory access (DMA) channel to allow for multiple data sources. Figure 2 shows CPU A and CPU B sharing one memory module. For instance, with the Interdata 8/32 fourteen ported memory module, one could have fourteen (in any combination of CPUs and external devices) accessing one shared memory module. Some manufacturers can hookup only CPUs to shared memory and not special external devices.

Perkin-Elmer uses the same hardware option (i.e., Local Memory Interface) to provide a parallel direct memory access path from disk to local memory for simultaneous transfer operations while data is transferred to the shared memory.

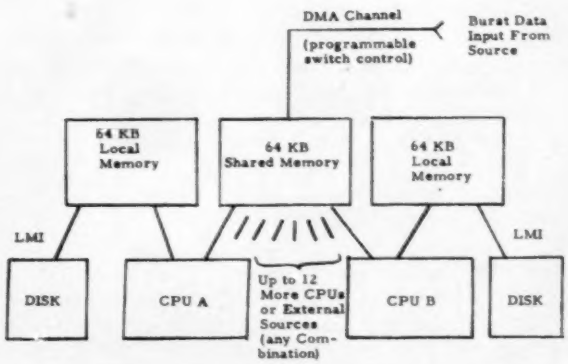


Figure 2—Perkin-Elmer's shared memory expansion capability

*Abbreviations used: DMA—direct memory access; LMI—local memory interface

In addition, Perkin-Elmer has multiple memory banking where each memory bank operates independently of the other banks and multiple CPUs have simultaneous access to the different banks in the system thus increasing the overall throughput. A shared memory module can be very large (e.g. Perkin-Elmer 16M Bytes) where the memory porting is to/from the memory controller, not the memory module itself. Perkin-Elmer 16M Bytes shared memory is to/from four memory controllers capable of looking like one fourteen ported shared memory.

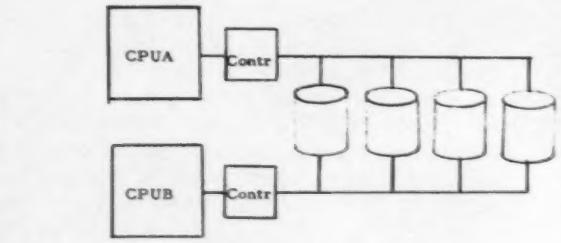
Any manufacturer offering multiport (i.e., shared) memory operations has to have a "test and set" instruction or the equivalent to facilitate accessing memory locations in a well disciplined manner and to preclude contention or conflicts with other CPUs accessing the same memory locations. In other words, the "test and set" instruction provides the necessary mechanism for software (i.e., program accessing memory) synchronization. For instance, the Perkin-Elmer's Test and Set (TS) instruction executes in only one instruction cycle and the built in hardware priority ensures that the other CPUs cannot interfere during the execution of the TS instruction. The TS instruction which executes within one instruction cycle first reads the first memory location in the buffer and writes a one in the significant bit position of the word to set a busy flag. If the busy flag has previously been set, then the condition code reflects this, and the CPU tries again later.

The other even more important aspect is how should contention amongst fourteen CPUs be handled in a well disciplined manner? There are several approaches to signal buffer ready or buffer available between CPUs. The simplest is by polling (i.e., testing if the flag is set or reset) the first buffer memory location with a test and set instruction periodically to determine if the buffer data is available for that CPU and then to proceed to read the buffer out. When only a few CPUs are on a shared memory, the controller allocates memory cycles in a round robin manner to access words. Another way supported by manufacturers is to use hardware interrupts. In this case, one CPU triggers an interrupt to another CPU when it has finished reading (i.e., relinquishing) the data buffer so the other CPU knows it can access the data buffer at that time. With a large number of CPUs such as fourteen, a priority scheme is used and the lowest priority job encounters some waits or delays. For instance Perkin-Elmer built into their memory controller both sequential access (i.e., round robin) and fixed priority with strapping to select one or the other. The third and most complicated way is to use some sort of data communication line which involves software message processing in effect from the CPU relinquishing the data buffer to the next CPU available to read the data buffer.

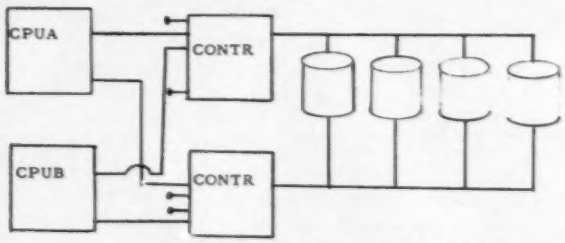
Shared disk configurations

Shared disk configurations are utilized to achieve significantly different benefits from those for which shared main memory configurations are implemented. One major application of shared disk configurations is to provide large file storage for long term sequential processing steps where data is collected and updated on demand. An obvious example of this is online airline reservation systems. Data is really not reduced as in shared main memory applications but in fact is a data base which requires continuous up-dates for 24-hour-a-day operations. If one briefly considers what capabilities and support are necessary to safely have more than one CPU accessing data base files, it turns out you probably need to modify the existing operating system to achieve this. For example, in a standard operating system, a program may open a file, allocate 1000 blocks and begin writing records. Typical operating systems only write an EOF (end-of-file) when a file is closed. To conserve disk I/O during "append" operations, the number of blocks used, and indicated in the header, will be updated only at "file closed" time. So there is a problem that when the disk is suddenly switched to a new computer it sees 1000 blocks allocated and 0 blocks used. The EOF is in the first computer and not the on-disk. Thus a significant amount of overhead occurs in shared disk systems since EOFs must be written each time. In addition, many file systems block records to conserve disk I/O. Thus a program may write a half dozen records and the operating system will buffer it in main memory until a seventh is written and then write the block to disk. If the disk is switched or the computer accessing disk changes prior to the block being written to disk, the records blocked in main memory are lost. Also any records crossing block boundaries would be written to the shared files in an uncontrolled order so for these reasons block buffering cannot be allowed.

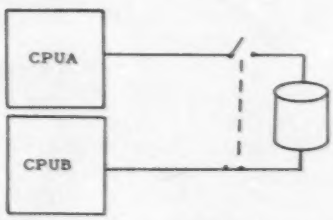
The benefit of shareable disk in the above case is to have reasonably fast response (i.e., high throughput) in accessing the data base, with power outages not causing data base to be lost. Even more popular, however, is higher availability because one computer going down does not bring the system down. Manufacturers now offer support with appropriate hardware (i.e., time-out logic) and operating systems to provide the necessary capability to accommodate switchover between CPUs in case one fails. This latter attribute is by far the most predominant capability now being sought by the user. Thus achievement of high reliability and graceful degradation involves some sort of shared or switchable disk capabilities. The distinction here is that shared disks are used in on-line transaction oriented systems whereas switchable disks are utilized in redundant or fail-over operations to facilitate graceful degradation of operations. Switchover can be done by manual operator control or under software control. These kinds of applications have typically used dual-ported and even tri-ported disk drives, which means separate disk controller paths exist to the disk drive electronics. In a fully redundant configuration, a failure in the CPU, channel, controller, or disk drive unit will not affect the proper operation of the other path. Note that dual port disk controllers (which accept data from either two different CPUs or from two channels of the same computer) can be a subset capability of shared disks but having dual ported disk controllers does not necessarily mean shared disk operations. The different types are shown in Figure 3.



(a) Shared (Dual Access) Disk Drive



(b) Dual/Multi-Ported Disk Controllers and dual access drives



(c) Switched (Bus Switch or Peripheral Switch) Disk System

Figure 3—Shared/switched disk configurations

RELATIVE COMPARISON OF NETWORK AND MULTIPROCESSING ARCHITECTURE TYPES

The network and multiprocessor implementation approaches are given in Table V. This is qualitative and not quantitative, although numbers or a range could be given for relevant user characteristics, such as average response time and average transfer rate. Each user has a different mix of criteria and weighting of each criterion, so it would be difficult to establish an overall rating system. What is useful, hopefully, is to identify the key performance advantage of each type and the inherent limitations or disadvantages of each type.

The distributed/network (Type 1) approach is available from several manufacturers where the user uses the off-the-shelf hardware/software support with no development required except to define the network configuration and resources and install the application software modules to do the user data processing functions. This approach gives the user off-the-shelf networking capability but at the expense of considerable general purpose capability that a user may only partially need. In addition, special applications do not use all the software support to handle their kinds of network structures. However, this capability minimizes required user's knowledge of the details at the expense of the overhead to provide all these conveniences. Users can now rapidly implement (at lower cost) distributed/network applications, but at the price of slower data transfer rates and longer response times than with their own network architectures. Also, it is usually difficult to change or modify generalized distributed/network software when required.

Multiple CPUs interconnected with some form of high speed bus structure is a logical way to achieve fast data transfer rates for multiprocessing system. These vary as to design implementation specifics and hardware component technologies that are employed to maximize throughput. Multiprocessing in turn provides the user with faster response times for processing tasks or jobs. The disadvantage is that in order to achieve this high throughput performance, the manufacturer or user must develop specialized software to tailor his application and to maximally utilize these capabilities. Only that manufacturer supports the hardware and software, and it has no systems compatibility with other manufacturers. All the multiprocessing systems are restricted to one location but can support terminals remotely over communication lines.

The major advantage of shared main memory via multiports is the almost instant accessibility by many CPUs to data where the processing time is much longer than the data collection time. Shared memory can thus be used to achieve extremely fast throughputs. It drastically reduces overhead in both timing and storage requirements by providing common access to many CPUs concurrently at semiconductor cycle times and by not having to duplicate copies of data in many memory banks. There is also a hardware cost savings in that shared memory architecture requires no channel hardware which is typically in short supply to support all the peripheral devices.

The last type of approach to linking together computers is a loosely coupled arrangement via the shared disk. This allows each computer to process jobs with no special timing or synchronization between machines. In other words, shared disk operations are asynchronous. For this reason the operating system has to be modified and the file manager system is impacted depending on the application and the file security and protection required. Shared disks typically use high speed direct memory access channels for reading and writing files, but, of course, this is slow compared to shared main memory access times. The disk, however, provides much larger storage capacities for two orders of magnitude lower cost per byte. Also disk data is nonvolatile when the power goes off which is a disadvantage of semiconductor shared main memory without batteries. Shared main memory supports as many CPUs as there are ports on the memory module, whereas dual and triported disks are about the practical limit as a shared resource. And last, but not least, just about every manufacturer in the business offers shared disk hardware capability.

TABLE V—Comparison of four computer configuration types

	Type 1 Distributed/Network	Type 2 Multiple CPUs Interconnected With Bus Hardware Configurations	Type 3 Shared Main Memory	Type 4 Shared Disk
ADVANTAGES	Several manufacturers offer complete off-the-shelf operating system support Distributed/network system easy to implement with no systems development just application software Some manufacturers support any network configuration (e.g., star, ring, n-level hierarchy) with both hardware and data communication links	Very fast transfer rates Number of CPUs limited only by adapter hardware design Processing response time is inherently faster than other types of configurations	Very fast and convenient access to data for many CPUs Number of CPUs limited only by the number of memory ports Saves by eliminating storage of common data Minimize input-output operations (i.e. no channels required)	Minimizes impact on computer operations especially timing Timing/synchronization requirements relatively easy to establish Shared disk uses direct memory access channels for high transfer rates Requires standard data base file/record formats between CPUs Most manufacturers offer this hardware capability Large capacity, non-volatile
DISADVANTAGES	Higher operating system overhead for generalized design Slower transfer rates and longer response times typically Difficult to change generalized design hardware and software	Usually not compatible with data communication link protocols Very specialized operating system software support Application flexibility is limited by software and hardware design system peculiar to the manufacturer offering this capability Uses hardware cables to interconnect and there are cable distance limitations	Special instructions and software support has to be used All installations have differences so manufacturer has only partial software support package Common data capacity limited to capacity of the main memory address structure Only a few manufacturers offer this capability as yet	Response time to access file is much slower 100 milliseconds versus 10 microseconds for shared main memory Disk storage overhead is high Limited to three CPU accessibility per shared disk Special software development expense

FUTURE TRENDS

We have every expectation that distributed network and multiprocessing types of architectural approaches will continue to increase in terms of the manufacturers offering this support and their use in application implementations. The reasons vary for why each type will evolve with more and more uses, and the trend seems clearly established already (we have only begun to see the tip of the iceberg). Distributed/network systems are clearly evident in all kinds of business operations with remote nodes feeding a hierarchy or a centralized node to process data orders, update inventory and provide all the many functions described in so many other articles. Their future looks boundless and unlimited.

As more users being to realize that multiprocessing capabilities exist and that one has only to take a sound, practical approach to implementing an architecture to map those resources to their application, the risk of multiprocessing will remain a memory to only the old timers in the business.

The high speed bus type of multiprocessing architecture which provides automatic fail-soft or fail-over capability will emerge into its own type and have its own marketplace for products to compete with large scale, single computer machines. Many users will want this because it offers significant cost advantages with low acquisition cost and no staff of systems programmers needed to support operation. The last type is shared disk system which will be used less for purely multiprocessing purposes because shared main memory costs are decreasing all the time and more for large data base applications. But use of shared or switchable disks will steadily increase to accommodate fail-soft or fail-over capability to provide users the high availability and reliability business demands.